What is claimed is:

1. An apparatus comprising:

a processor executive (PE) to handle an operating system executive (OSE) in a secure environment, the secure environment having a fused key (FK) and associated with an isolated memory area in a platform, the OSE to manage a subset of an operating system (OS) running on the platform, the platform having a processor operating in one of a normal execution mode and an isolated execution mode, the isolated memory area being accessible to the processor in the isolated execution mode;

a PE supplement to supplement the PE with a PE manifest representing the PE and a PE identifier to identify the PE; and

a PE handler to handle the PE using the FK and the PE supplement.

2. The apparatus of claim 1 further comprises:

a boot-up code to boot up the platform following a power on.

3. The apparatus of claim 2 wherein the secure environment includes an OSE supplement to supplement the OSE with an OSE manifest representing the OSE and an OSE identifier to identify the OSE.

4. The apparatus of claim 3 wherein the PE handler comprises:

a PE loader to load the PE and the PE supplement from a PE memory into the isolated memory area using a parameter block provided by the boot-up code;

a PE manifest verifier to verify the PE manifest; and

a PE verifier to verify the PE using the PE manifest and a constant derived from the FK.

5. The apparatus of claim 4 wherein the PE handler further comprises:

a PE key generator to generate a PE key using the FK;

a PE identifier logger to log the PE identifier in a storage; and

a PE entrance/exit handler to handle a PE entry and a PE exit.

6. The apparatus of claim 5 wherein the PE key generator comprises:

a PE key combiner to combine the PE identifier and the FK, the combined PE identifier and the FK corresponding to the PE key.

7. The apparatus of claim 6 wherein the PE comprises:

an OSE loader to load the OSE and the OSE supplement into the isolated memory area;

an OSE manifest verifier to verify the OSE manifest; and

an OSE verifier to verify the OSE.

8. The apparatus of claim 7 wherein the PE further comprises:

an OSE key generator to generate an OSE key;

an OSE identifier logger to log the OSE identifier in a storage; and

an OSE entrance/exit handler to handle an OSE entry and an OSE exit.

9. The apparatus of claim 8 wherein the OSE key generator comprises:

a binding key generator to generate a binding key (BK) using the PE key; and

an OSE key combiner to combine the OSE identifier and the BK, the combined OSE identifier and the BK corresponding to the OSE key.

10. The apparatus of claim 9 wherein the OSE comprises:

a module loader to load a module into the isolated memory area;

a page manager to manage paging in the isolated memory area; and

an interface handler to handle interface with the OS.

11. The apparatus of claim 9 wherein the module is one of an application module, an applet module, and a support module.

12. The apparatus of claim 11 wherein the OSE further comprises:

an applet key generator to generate an applet key associating with the applet module.

13. The apparatus of claim 12 wherein the applet key generator comprises:

an applet key combiner to combine the OSE key with an applet identifier identifying the applet module, the combined OSE key and the applet identifier corresponding to the applet key.

14. The apparatus of claim 13 wherein the boot up code comprises:

a PE locator to locate the PE and the PE supplement, the PE locator transferring the PE and the PE supplement into the PE memory at a PE address;

a PE recorder to record the PE address in the parameter block; and

an instruction invoker to execute an isolated create instruction, the isolated create instruction loading the PE handler into the isolated memory area.

15. The apparatus of claim 14 wherein the isolated create instruction performs an atomic sequence, the atomic sequence being non-interruptible.

16. The apparatus of claim 15 wherein the atomic sequence comprises:

a physical memory operation to verify if the processor is in a flat physical page mode;

an atomic read-and-increment operation to read and increment a thread count register in a chipset, the read-and-increment operation determining if the processor is the first processor in the isolated execution mode;

an isolated memory area control operation to configure the chipset using a configuration storage;

a processor isolated execution operation to configure the processor in the isolated execution mode; and

an PE handler loading operation to load the PE handler into the isolated memory area.

17. The apparatus of claim 16 wherein the atomic sequence further comprises:

a PE handler verification to verify the loaded PE handler; and

an exit operation to transfer control to the loaded PE handler.

18. The apparatus of claim 16 wherein the processor isolated execution operation comprises:

a chipset read operation to read the configuration storage in the chipset when the processor is not a first processor in the isolated execution mode; and

a processor configuration operation to configure the processor according to the configuration storage in the chipset when the processor is not the first processor in the isolated execution mode.

19. The apparatus of claim 18 wherein the chipset includes at least one of a memory controller hub (MCH) and an input/output controller hub (ICH).

20. The apparatus of claim 8 wherein the storage is in an input/output controller hub (ICH) external to the processor.

21. A method comprising:

handling an operating system executive (OSE) by a processor executive (PE) in a secure environment, the secure environment having a fused key (FK) and associated with an isolated memory area in a platform, the OSE to manage a subset of an operating system (OS) running on the platform, the platform having a processor operating in one of a normal execution mode and an isolated execution mode, the isolated memory area being accessible to the processor in the isolated execution mode;

supplementing the PE using a PE supplement, the PE supplement having a PE manifest representing the PE and a PE identifier to identify the PE; and

handling the PE by a PE handler using the FK and the PE supplement.

22. The method of claim 21 further comprises:

booting up the platform by a boot-up code following a power on.

23. The method of claim 22 wherein the secure environment includes an OSE supplement to supplement the OSE with an OSE manifest representing the OSE and an OSE identifier to identify the OSE.

24. The method of claim 23 wherein handling the PE comprises:

loading the PE and the PE supplement from a PE memory into the isolated memory area using a parameter block provided by the boot-up code;

verifying the PE manifest; and

verifying the PE using the PE manifest and a constant derived from the FK.

25. The method of claim 24 wherein handling the PE further comprises:

generating a PE key using the FK;

logging the PE identifier in a storage; and

handling a PE entry and a PE exit.

26. The method of claim 25 wherein generating the PE key comprises:

combining the PE identifier and the FK, the combined PE identifier and the FK corresponding to the PE key.

27. The method of claim 26 wherein handling the OSE comprises:

loading the OSE and the OSE supplement into the isolated memory area;

verifying the OSE manifest; and

verifying the OSE.

28. The method of claim 27 wherein handling the OSE further comprises:

generating an OSE key;

logging the OSE identifier in a storage; and

handling an OSE entry and an OSE exit.

29. The method of claim 28 wherein generating the OSE key comprises:

generating a binding key (BK) using the PE key; and

combining the OSE identifier and the BK, the combined OSE identifier and the BK corresponding to the OSE key.

30. The method of claim 29 wherein managing the subset of the OS comprises:

loading a module into the isolated memory area;

managing paging in the isolated memory area; and

handling interface with the OS.

31. The method of claim 29 wherein the module is one of an application module, an applet module, and a support module.

32. The method of claim 31 wherein managing the subset of the OS further comprises:

generating an applet key associating with the applet module.

33. The method of claim 32 wherein generating the applet key comprises:

combining the OSE key with an applet identifier identifying the applet module, the combined OSE key and the applet identifier corresponding to the applet key.

34. The method of claim 33 wherein booting up comprises:

locating the PE and the PE supplement;

transferring the PE and the PE supplement into the PE memory at a PE address;

recording the PE address in the parameter block; and

executing an isolated create instruction, the isolated create instruction loading the PE handler into the isolated memory area.

35. The method of claim 34 wherein executing the isolated create instruction comprises performing an atomic sequence, the atomic sequence being non-interruptible.

36. The method of claim 35 wherein performing the atomic sequence comprises:

verifying if the processor is in a flat physical page mode;

reading and incrementing a thread count register in a chipset to determine if the processor is the first processor in the isolated execution mode;

configuring the chipset using a configuration storage;

configuring the processor in the isolated execution mode; and

loading the PE handler into the isolated memory area.

37. The method of claim 36 wherein performing the atomic sequence further comprises:

verifying the loaded PE handler;

transferring control to the loaded PE handler.

38. The method of claim 36 wherein configuring the processor in the isolated execution mode comprises:

reading the configuration storage in the chipset when the processor is not a first processor in the isolated execution mode; and

configuring the processor according to the configuration storage in the chipset when the processor is not the first processor in the isolated execution mode.

39. The method of claim 38 wherein the chipset includes at least one of a memory controller hub (MCH) and an input/output controller hub (ICH).

40. The method of claim 28 wherein the storage is in an input/output controller hub (ICH) external to the processor.

41. A computer program product comprising:

a machine useable medium having computer program code embedded therein, the computer program product having:

computer readable program code for handling an operating system executive (OSE) by a processor executive (PE) in a secure environment, the secure environment having a fused key (FK) and associated with an isolated memory area in a platform, the OSE to manage a subset of an operating system (OS) running on the platform, the platform having a processor operating in one of a normal execution mode and an isolated execution mode, the isolated memory area being accessible to the processor in the isolated execution mode;

computer readable program code for supplementing the PE using a PE supplement, the PE supplement having a PE manifest representing the PE and a PE identifier to identify the PE; and

computer readable program code for handling the PE by a PE handler using the FK and the PE supplement.

42. The computer program product of claim 41 further comprises:

computer readable program code for booting up the platform by a boot-up code following a power on.

43. The computer program product of claim 42 wherein the secure environment includes an OSE supplement to supplement the OSE with an OSE manifest representing the OSE and an OSE identifier to identify the OSE.

44. The computer program product of claim 43 wherein the computer readable program code for handling the PE comprises:

computer readable program code for loading the PE and the PE supplement from a PE memory into the isolated memory area using a parameter block provided by the boot-up code;

computer readable program code for verifying the PE manifest; and

computer readable program code for verifying the PE using the PE manifest and a constant derived from the FK.

45. The computer program product of claim 44 wherein the computer readable program code for handling the PE further comprises:

computer readable program code for generating a PE key using the FK;

computer readable program code for logging the PE identifier in a storage; and

computer readable program code for handling a PE entry and a PE exit.

46. The computer program product of claim 45 wherein the computer readable program code for generating the PE key comprises:

computer readable program code for combining the PE identifier and the FK, the combined PE identifier and the FK corresponding to the PE key.

47. The computer program product of claim 46 wherein the computer readable program code for handling the OSE comprises:

computer readable program code for loading the OSE and the OSE supplement into the isolated memory area;

computer readable program code for verifying the OSE manifest; and

computer readable program code for verifying the OSE.

48. The computer program product of claim 47 wherein the computer readable program code for handling the OSE further comprises:

computer readable program code for generating an OSE key;

computer readable program code for logging the OSE identifier in a storage; and

computer readable program code for handling an OSE entry and an OSE exit.

49. The computer program product of claim 48 wherein the computer readable program code for generating the OSE key comprises:

computer readable program code for generating a binding key (BK) using the PE key; and

computer readable program code for combining the OSE identifier and the BK, the combined OSE identifier and the BK corresponding to the OSE key.

50. The computer program product of claim 49 wherein the computer readable program code for managing the subset of the OS comprises:

computer readable program code for loading a module into the isolated memory area;

computer readable program code for managing paging in the isolated memory area; and

computer readable program code for handling interface with the OS.

51. The computer program product of claim 49 wherein the module is one of an application module, an applet module, and a support module.

52. The computer program product of claim 51 wherein the computer readable program code for managing the subset of the OS further comprises:

computer readable program code for generating an applet key associating with the applet module.

53. The computer program product of claim 52 wherein the computer readable program code for generating the applet key comprises:

computer readable program code for combining the OSE key with an applet identifier identifying the applet module, the combined OSE key and the applet identifier corresponding to the applet key.

54. The computer program product of claim 53 wherein the computer readable program code for booting up comprises:

computer readable program code for locating the PE and the PE supplement;

computer readable program code for transferring the PE and the PE supplement into the PE memory at a PE address;

computer readable program code for recording the PE address in the parameter block; and

computer readable program code for executing an isolated create instruction, the isolated create instruction loading the PE handler into the isolated memory area.

55. The computer program product of claim 54 wherein the computer readable program code for executing the isolated create instruction comprises computer readable program code for performing an atomic sequence, the atomic sequence being non-interruptible.

56. The computer program product of claim 55 wherein the computer readable program code for performing the atomic sequence comprises:

computer readable program code for verifying if the processor is in a flat physical page mode;

computer readable program code for reading and incrementing a thread count register in a chipset to determine if the processor is the first processor in the isolated execution mode;

computer readable program code for configuring the chipset using a configuration storage;

computer readable program code for configuring the processor in the isolated execution mode; and

computer readable program code for loading the PE handler into the isolated memory area.

57. The computer program product of claim 56 wherein the computer readable program code for performing the atomic sequence further comprises:

computer readable program code for verifying the loaded PE handler;

computer readable program code for transferring control to the loaded PE handler.

58. The computer program product of claim 56 wherein the computer readable program code for configuring the processor in the isolated execution mode comprises:

computer readable program code for reading the configuration storage in the chipset when the processor is not a first processor in the isolated execution mode; and

computer readable program code for configuring the processor according to the configuration storage in the chipset when the processor is not the first processor in the isolated execution mode.

59. The computer program product of claim 58 wherein the chipset includes at least one of a memory controller hub (MCH) and an input/output controller hub (ICH).

60. The computer program product of claim 48 wherein the storage is in an input/output controller hub (ICH) external to the processor.

61. A system comprising:

a processor operating in one of a normal execution mode and an isolated execution mode;

a memory coupled to the processor having an isolated memory area accessible to the processor in the isolated execution mode; and

an executive subsystem comprising:

a processor executive (PE) to handle an operating system executive (OSE) in a secure environment, the secure environment having a fused key (FK) and associated with the isolated memory, the OSE to manage a subset of an operating system (OS),

a PE supplement to supplement the PE with a PE manifest representing the PE and a PE identifier to identify the PE, and

a PE handler to handle the PE using the FK and the PE supplement.

62. The system of claim 61 wherein the executive subsystem further comprises:

a boot-up code to boot up the platform following a power on.

63. The system of claim 62 wherein the secure environment includes an OSE supplement to supplement the OSE with an OSE manifest representing the OSE and an OSE identifier to identify the OSE.

64. The system of claim 63 wherein the PE handler comprises:

a PE loader to load the PE and the PE supplement from a PE memory into the isolated memory area using a parameter block provided by the boot-up code;

a PE manifest verifier to verify the PE manifest; and

a PE verifier to verify the PE using the PE manifest and a constant derived from the FK.

65. The system of claim 64 wherein the PE handler further comprises:

a PE key generator to generate a PE key using the FK;

a PE identifier logger to log the PE identifier in a storage; and

a PE entrance/exit handler to handle a PE entry and a PE exit.

66. The system of claim 65 wherein the PE key generator comprises:

a PE key combiner to combine the PE identifier and the FK, the combined PE identifier and the FK corresponding to the PE key.

67. The system of claim 66 wherein the PE comprises:

an OSE loader to load the OSE and the OSE supplement into the isolated memory area;

an OSE manifest verifier to verify the OSE manifest; and

an OSE verifier to verify the OSE.

68. The system of claim 67 wherein the PE further comprises:

an OSE key generator to generate an OSE key;

an OSE identifier logger to log the OSE identifier in a storage; and

an OSE entrance/exit handler to handle an OSE entry and an OSE exit.

69. The system of claim 68 wherein the OSE key generator comprises:

a binding key generator to generate a binding key (BK) using the PE key; and

an OSE key combiner to combine the OSE identifier and the BK, the combined OSE identifier and the BK corresponding to the OSE key.

70. The system of claim 69 wherein the OSE comprises:

a module loader to load a module into the isolated memory area;

a page manager to manage paging in the isolated memory area; and

an interface handler to handle interface with the OS.

71. The system of claim 69 wherein the module is one of an application module, an applet module, and a support module.

72. The system of claim 71 wherein the OSE further comprises:

an applet key generator to generate an applet key associating with the applet module.

73. The system of claim 72 wherein the applet key generator comprises:

an applet key combiner to combine the OSE key with an applet identifier identifying the applet module, the combined OSE key and the applet identifier corresponding to the applet key.

74. The system of claim 73 wherein the boot up code comprises:

a PE locator to locate the PE and the PE supplement, the PE locator transferring the PE and the PE supplement into the PE memory at a PE address;

a PE recorder to record the PE address in the parameter block; and

an instruction invoker to execute an isolated create instruction, the isolated create instruction loading the PE handler into the isolated memory area.

75. The system of claim 74 wherein the isolated create instruction performs an atomic sequence, the atomic sequence being non-interruptible.

76. The system of claim 75 wherein the atomic sequence comprises:

a physical memory operation to verify if the processor is in a flat physical page mode;

an atomic read-and-increment operation to read and increment a thread count register in a chipset, the read-and-increment operation determining if the processor is the first processor in the isolated execution mode;

an isolated memory area control operation to configure the chipset using a configuration storage;

a processor isolated execution operation to configure the processor in the isolated execution mode; and

an PE handler loading operation to load the PE handler into the isolated memory area.

77. The system of claim 76 wherein the atomic sequence further comprises:

a PE handler verification to verify the loaded PE handler; and

an exit operation to transfer control to the loaded PE handler.

78. The system of claim 76 wherein the processor isolated execution operation comprises:

a chipset read operation to read the configuration storage in the chipset when the processor is not a first processor in the isolated execution mode; and

a processor configuration operation to configure the processor according to the configuration storage in the chipset when the processor is not the first processor in the isolated execution mode.

79. The system of claim 78 wherein the chipset includes at least one of a memory controller hub (MCH) and an input/output controller hub (ICH).

80. The system of claim 68 wherein the storage is in an input/output controller hub (ICH) external to the processor.